A Novel Partition Method for Busy Urban Area Based on Spatial-Temporal Information

Zhengyang Ai[1,3], Kai Zhang[2], Shupeng Wang[1†], Chao Li[2†], Xiao-yu Zhang[1], and Shicong Li[2]

[1] Institute of Information Engineering, Chinese Academy of Sciences, Beijing, China
{ aizhengyang, wangshupeng, zhangxiaoyu }@iie.ac.cn
[2] National Computer Network Emergency Response Technical Team/Coordination Center of China, Beijing, China
{ zhangkai, lichao, lishicong }@cert.org.cn
[3] School of Cyber Security, University of Chinese Academy of Sciences, Beijing, China

Abstract. Finding the regions where people appear plays a key role in many fields like user behavior analysis, urban planning, etc. Therefore, how to partition the world, especially the urban areas where people are crowd and active, into regions is very crucial. In this paper, we propose a novel method called Restricted Spatial-Temporal DBSCAN (RST-DBSCAN). The key idea is to partition busy urban areas based on spatial-temporal information. Arbitrary and separated shapes of regions in urban areas would be then obtained. Besides, we would further get busier region earlier by RST-DBSCAN. Experimental results show that our approach yields significant improvements over existing methods on a real-world dataset extracted from Gowalla, a location-based social network.

Keywords: Area partition, density-based clustering, social link mining, location-based social network.

1 Introduction

With the rapid development of internet, cyberspace has become an important field for entertainment, consumption, etc. If we associate the cyberspace with geospatial, i.e. linking the online with offline, a mass of user behaviors could be mined, which would be of great help for recommendation system, friendship prediction, urban planning, etc. [1]. The first step for mapping cyberspace to geospatial is to find the regions where users locate. Therefore, how to divide the world, especially urban areas, becomes an important and necessary intermediate link. Traditionally, three methods are widely used for area partition, i.e. **address method**, **rigid method** and **cluster method**. However, these methods all have limitations, which would be introduced in detail as follows.

Address method refers to partitioning areas with physical address, like Starbucks, KFC, etc. In fact, as addresses often reveal user behaviors, like shopping, having dinner, entertainment, etc., this method is commonly used to study either user behavior prediction or product recommendation [2–4]. Although achieving high precision, address

[†] Corresponding author.

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22741-8_17

method has limitation in the scale of regions. And it's also too difficult for the method to find interactions between two users. For instance, the sparsity of user interactions in Foursquare and Yelp are greater than 99.99% [5]. As a result, it is inappropriate to do researches related to user interactions using this method.

Grid method refers to partitioning areas with gird. The metric for partitioning areas is usually kilometer or degree of latitude and longitude. For instance, L. Backstrom, et.al. divided the US into 0.01×0.01 degree regions (about 0.4 square miles) [6], while Cho E, et.al discretized the world into 25×25km cells [7]. This partitioning method is simple and practicable, and obtained regions could cover an ideal range to find interactions between users. Therefore, it is often used to study user social relationships [8–10]. However, its disadvantages are also obvious. The method could not partition areas based on practical significance or independent meaning, and always divides a function unit, e.g. a shop or a store, into two or more separate girds, missing the interaction information of users. Besides, it is hard to set a proper size for girds, since little size may miss interactions, while big size may cover too many user traces and make the extraction of interactions too hard to fulfill.

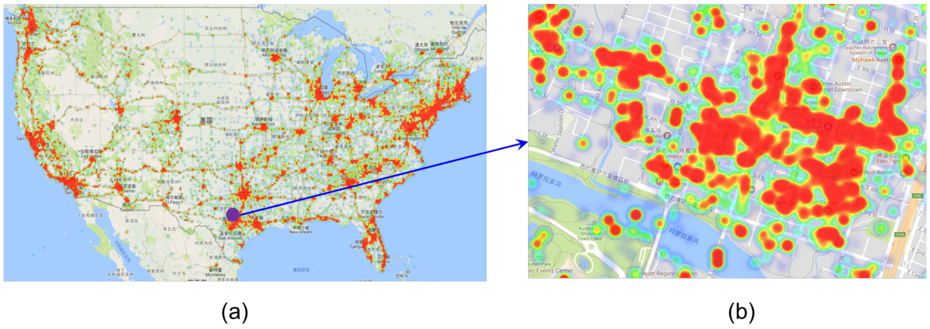
Cluster method refers to partitioning areas by clustering locations in the form of GPS, and density-based clustering (DBSCAN) is a representative method. C. Zhou, et.al designed a new approach based on DBSCAN, with which arbitrary shapes could be obtained, and areas could be partitioned according to practical significance or independent meaning as a result [11]. Besides, reasonable parameter value is easy to set for DBSCAN when partitioning areas. By this way, DBSCAN outperforms other cluster methods evidently, and is widely used later when dealing with personal locations [12, 13]. However, as DBSCAN works based on density, a cluster would "spread" infinitely when density meets the requirement. As a result, intensive locations in busy urban areas would be clustered into a same group with DBSCAN. Fig. 1 shows the hot map of user check-ins in the busy area of Austin, US with the dataset extracted from Gowalla, a location-based social network. Intensive check-ins and locations as shown in the figure, make it difficult to partition the area with DBSCAN. In fact, this area is taken as a region when clustered with 0.001 degree. Therefore, DBSCAN doesn't work well when dealing with a mass of locations, especially with those in busy urban areas.

In this paper, we propose a method for partitioning busy urban areas with spatial-temporal information in LBSN (Location-Based Social Network). On one hand, arbitrary shapes would be obtained by designing method with density-based clustering, overcoming the stiff of grid method. On the other hand, a boundary is designed ingeniously for limiting the scale of clusters, overcoming the infinite "spread" of DBSCAN. Our approach is validated using real user data and show a good performance.

Our contributions are concluded as follows:

- We propose a new method, **RST-DBSCAN**, for partitioning busy urban areas based on spatial-temporal information, with which arbitrary and separated shapes of regions would be obtained reasonably;
- We propose a new time mapping method to connect temporal information with spatial information. The method makes it possible to partition areas with 3-dimensional spatial-temporal information;

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22741-8_17



(a) (b)

Fig. 1: The hot map of users' check-ins in the busy urban area of Austin, US, around the Texas government.

- We further propose to cluster from the locations which own greater density. In this way, we can get busier regions earlier, making the approach more reasonably;
- We visually and quantitatively evaluate our approach with a location-based social network, Gowalla. Results show that the performance of RST-DBSCAN outperforms that of competitors.

The rest of this paper is organized as follows. Section 2 introduces the design of RST-DBSCAN. Section 3 describes experiments. Finally, section 4 gives the conclusion.

2 Design of the model

In this section, we introduce our method in the following steps. Definitions for designing RST-DBSCAN would be introduced at first. Then we introduce the ideas for designing RST-DBSCAN in the view of space and time respectively. Finally, we describe the method with schematic and pseudocode.

2.1 Definitions

DBSCAN is a well-known cluster method and works by clustering points which satisfy the density conditions into a group. Therefore, two important concepts, ε, the radius of a circle, and *MinPts*, the minimum number of neighbor points within that circle, are needed [14]. In this section, points are employed to denote locations in 2-dimensional spatial space. Then the circle with radius ε (named **neighbor region** here) means the scale covered by a location, and *MinPts* means the minimum number of location's neighbors. Besides, we make definitions for designing RST-DBSCAN as follows.

Definition1: Candidate core location

If the neighbor region of a location, p, covers at least *MinPts* locations, p is a candidate core location.

Definition2: Neighbor location

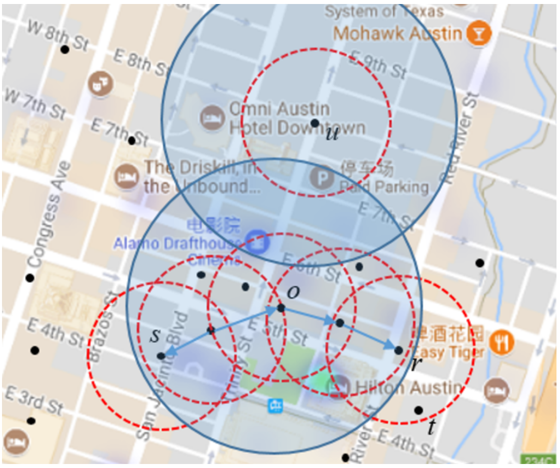


Fig. 2: The schematic of RST-DBSCAN

The neighbors of p, denoted by $N_\varepsilon(p)$, is defined by

$$N_\varepsilon(p) = \{q \in S \mid dist(p,q) \leq \varepsilon\} \tag{1}$$

Here, S is the set of all locations, and q is any location in the set.

Definition3: Core location

The location which starts a clustering process is taken as core location.

Definition4: Location directly density reachable

For a core location p and a location q, we say that q is directly density reachable from p if q is the neighbor location of p.

Definition5: Location density reachable

For a location p and a location q, we say q is density reachable from p if there is a chain of locations $q_1, q_2, ..., q_n$, such that $q_1 = q, q_n = p$, and q_{i+1} is directly density reachable from q_i. Here $1 \leq i \leq n$.

Definition6: Core-related location

Assume location n is density reachable from core location p and satisfies the following condition

$$\{n \in S \mid dist(p,n) \leq \lambda \cdot \varepsilon\} \tag{2}$$

Then location n is a core-related location for p. Besides, the circle region with radius $\lambda \cdot \varepsilon$ is called as **core-related region** of p.

What needs to note is that once a candidate core location is covered by a core location, it becomes a core-related location.

2.2 Area partition with spatial information.

For the sake of understanding, we introduce how RST-DBSCAN works with spatial information at first in this section, and then the application of time information would be introduced in the next section.

The schematic of RST-DBSCAN with $\lambda = 2$ is shown in Fig. 2. Here black points denote locations, circles of dotted lines denote the neighbor regions, and the circles of solid line denote core-related regions. RST-DBSCAN decides the core locations based on the number of neighbor locations. As the density around location o is the greatest, we take it as the first core location. Location s, r and t are all density reachable from o, so these locations belong to the same group when clustering with DBSCAN. However, as location t beyond the range of o's core-related region, it would not belong to the group if clustered by RST-DBSCAN. With core-related regions as the restricted condition, we would obtain separated regions in busy urban areas.

2.3 Area partition with temporal information

In section 2.2, we partition busy urban area by restricted DBSCAN with spatial information in the form of GPS. With the help of temporal information, we could continuously process urban area further. Although intensive check-in is an important feature in busy urban area, the intensity varies a lot over time. Fig. 3 shows the hot map of users' check-ins around the Texas government at different time periods. As can be seen, most of check-ins appear at commercial districts in the daytime, while most of them appear at resident area at night. In more detail, Fig. 4 shows the situation of check-ins in a few blocks away. As can be seen, check-ins are active in shopping mall in the afternoon and at dusk, while more users appear in bars at early morning. Therefore, conclusions can be drawn that users check-ins also reflect their living habits, and area partitioned with temporal information would contain more information.

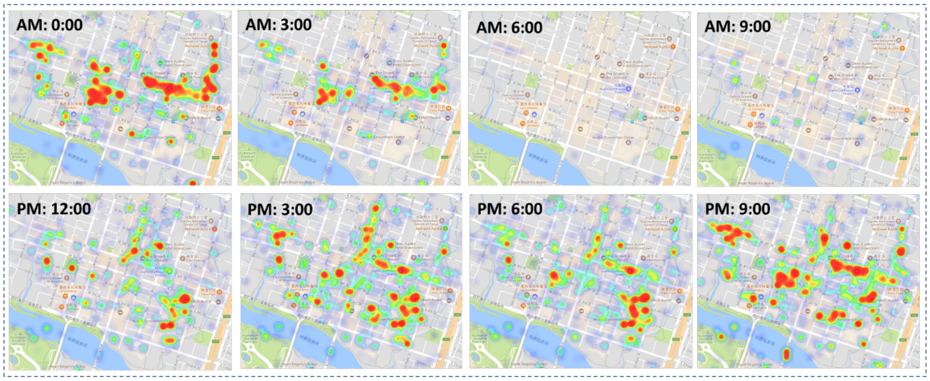


Fig. 3: The hot map of users' check-ins around the Texas government at different time periods

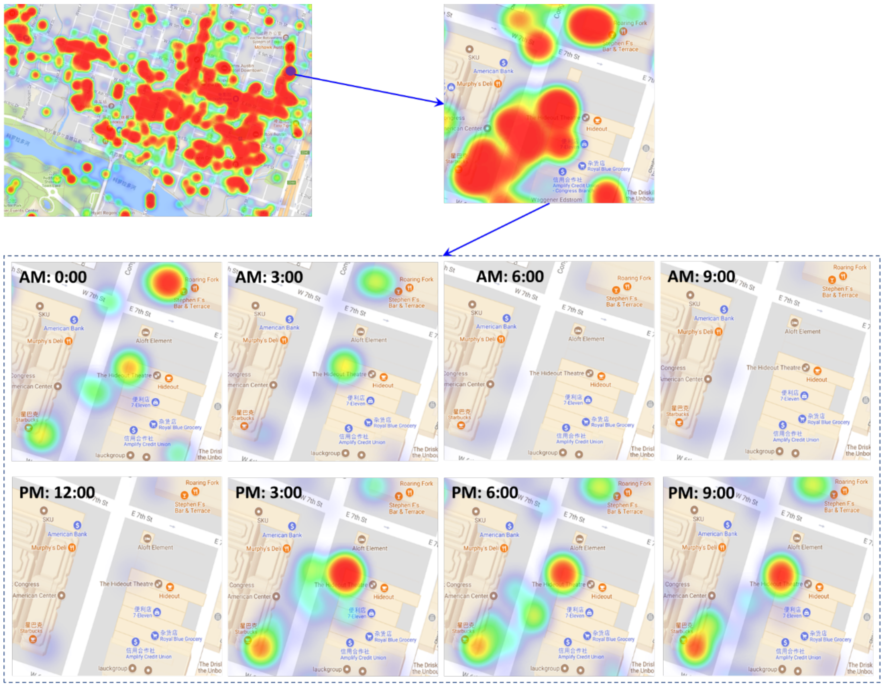


Fig. 4: The hot map of users' check-ins in a few blocks away at different time periods.

However, how to use temporal information for partitioning area is still a challenging issue. Previous studies often partition time with hour periods, which is not reasonable enough. For instance, a restaurant is active from 5:10 PM to 20:45 PM, and it is hard to partition this time period with traditional method. Therefore, we bring out a new time-mapping method to connect temporal information with spatial information. The formula of time-mapping is

$$l_{Time} = \vartheta \cdot \varepsilon \cdot (T_{hour} + \frac{T_{min}}{60} + \frac{T_{min}}{3600}) \tag{3}$$

Here ϑ is time-mapping parameter. When $\vartheta = 1$, 1 hour corresponds to ε degree in the spatial scale. Then we could deal with temporal information with RST-DBSCAN, with which arbitrary and reasonable time period would be obtained.

2.4 Description of the model

In this section, we would introduce the design of our approach. The pseudocode of RST-DBSCAN is shown in Algorithm 1. We take the check-in dataset D, the radius of a circle ε and the minimum number of neighbors *MinPts* as input, and obtain location clusters with the algorithm. Here D includes user coordinate of latitude and longitude and check-in time. Specific implementation details of RST-DBSCAN algorithm are as follows.

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22741-8_17

Algorithm 1: RST-DBSCAN

```
Input:
$\varepsilon$; MinPts;$\vartheta$; Dataset $D$
Output:
Location clusters;
• Initialization
$\Omega' = \Phi$;
$k = 0$;
$\Omega =<>$;
$l_{Time}= \vartheta \cdot \varepsilon \cdot (T_{hour} + \frac{T_{min}}{60} + \frac{T_{min}}{3600})$;
$\Gamma = \{D, l_{Time}\}$;
• Obtain the set of candidate core locations
for $j = 1, 2, \ldots, m$ do
    if $|N_\varepsilon(x_j)| \geq MinPts$ then
        $\Omega' = \Omega' \cup x_j$ ;
• Sort candidate core locations with number of neighbor locations
while $\Omega' \neq \Phi$ do
    select $p$ from $\Omega'$, st. $N_\varepsilon(p)$is the max in $\Omega'$;
    add $p$ to $\Omega$;
    $\Omega' = \Omega' \backslash p$;
• Obtain target clusters
while $\Omega \neq \Phi$ do
    $\Gamma_{old} = \Gamma$;
    $Q =< o >$, st. $o$ is the first location in $\Omega$;
    $\Gamma = \Gamma \backslash \{o\}$;
    while $Q \neq \Phi$ do
        select the first location $q$ in $Q$;
        if $|N_\varepsilon(q)| \geq MinPts$ then
            $\Delta = N_\varepsilon(q) \cap \Gamma$;
            while unprocessed location exists in $\Delta$ do
                select location $r$ in $\Delta$ randomly;
                $dist(r, o) = |r - o|$;
                if $dist(r, q) > \lambda * \varepsilon$ then
                    $\Delta = \Delta \backslash r$;
            add locations in $\Delta$ to $Q$;
            $\Gamma = \Gamma \backslash \Delta$;
    $C_k = \Gamma_{old} \backslash \Gamma$;
    $k = k + 1$;
    $\Omega = \Omega \backslash C_k$;
```

(1) Initialization

Set the number of clusters as 0. Compute the mapping result of time and obtain 3-dimensional dataset with unprocessed locations, Γ. The dataset of candidate core locations $\Omega'$and a queue Ω are initialized as empty, respectively. Here we say a location has been processed if it has been clustered into to a group, unprocessed otherwise.

(2) Obtain the set of candidate core locations

Traverse all the locations. Take location x for example. Calculate the number of x's neighbors, $|N_{\varepsilon}(x)|$, and then confirm whether x belongs to $\varOmega'$ or not according to the numerical value of $|N_{\varepsilon}(x)|$ and $MinPts$.

(3) Sort locations with their number of neighbors

Sort locations in $\varOmega'$ in descending order, according to their number of neighbors. Then a queue of candidate core locations $\varOmega$ is formed. Location with more neighbors in $\varOmega$ would be in more forward position.

(4) Obtain target clusters

Extract the first candidate core location q in $\varOmega$. Drop q if it has been processed, otherwise set it as the core location. Find all locations which are unprocessed and density reachable from q, and remove those that beyond the scope of q' core-related region. Then we would obtain a cluster with q as the center. Mark these locations as processed. Repeat the process above until $\varOmega$ is empty, and all locations will be clustered.

With above steps, a region where locations belong to the same cluster is formed at last. Based on practical significance and independent meaning, RST-DBSCAN could partition the area into arbitrary and separated shapes of regions.

3 Experiments

In this section, we carry on two experiments to verify the effectiveness of our approach. One is to partition a busy urban area with **RST-DBSCAN**. The other is to mine social links with user interactions in the regions obtained by RST-DBSCAN, as area partition plays an important role on social link mining.

3.1 Evaluation index

We adopt three performance metrices, precision (P), recall (R), F1-measure (F_1) to estimate the model, which can be calculated as follows. Let TP, TN, FP and FN denote the numbers of true positives, true negatives, false positives and false negatives respectively.

$$P = \frac{TP}{TP + FP} \tag{4}$$

$$R = \frac{TP}{TP + FN} \tag{5}$$

$$F_1 = 2\frac{P * R}{P + R} \tag{6}$$

As can be seen, F1-measure is the comprehensive value of precision and recall. Besides, links in social network are typically sparse, so F1-measure plays a more important role than accuracy when evaluating model. As a result, F1-measure is taken as a main performance index here.

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22741-8_17

Table 1: Comparison of DBSCAN and RST-DBSCAN when the value of ε varies.

ε	Region number/DBSCAN	Region number/RST-DBSCAN
0.005	1	4
0.001	1	14
0.0005	12	45

3.2 Experiment 1

Dataset

Experiments are performed on a publicly available dataset, which is extracted from a location-based social network namely Gowalla. It collects user check-ins and social links from 2009.2 to 2010.10 [7]. As most check-ins appear in Austin, US, we choose the urban area of Austin, around the government of Texas, as the target. The detailed scope is $30.262°$ N -$30.270°$ N, $97.730°$ W-$97.747°$W, and 5,173 users check in 72,131 times at 1,450 different locations.

Comparative approach

DBSCAN is taken as the comparative approach in this part.

Experimental setup

We set $MinPts = 1$ for both DBSCAN and RST-DBSCAN. λ is set to 3 for RST-DBSCAN. ε is set in the range of $0.005, 0.001, 0.0005$.

Evaluation results

When the value of ε varies, the results are shown in Table 1. As can be seen, for $\varepsilon = 0.005$ and 0.001, DBSCAN couldn't partition this area, while RST-DBSCAN divides the area into 4 and 14 regions respectively. For $\varepsilon = 0.005$, the area could be partitioned by DBSCAN, while the number is still far less than that by RST-DBSCAN. In fact, as 0.001 degree is roughly equal to 0.1 kilometers, smaller size even couldn't cover a unit like a store or a restaurant. Therefore, 0.001 is nearly the minimum size for rationality, and our approach outperforms DBSCAN substantially.

More visually, Fig. 5 shows 1,450 different locations in this area, represented by red points. When $\varepsilon = 0.001$, these locations belong to a same group clustered by DBSCAN. When clustered with RST-DBSCAN, the area is divided into 14 regions and results are shown in Fig. 6. Here we mark the locations with 14 different colors, and locations with the same color belong to the same group. We draw borders of these groups manually for viewing convenience.

3.3 Experiment 2

Dataset

The dataset extracted from Gowalla is also employed in this experiment. As inadequate information would make adverse effects on the experiment, we select users whose check-ins and friends are all at the top 2% (i.e. the time of check-ins is above 186, and the number of friends is above 52). There are 761 users which are matched with the conditions and there exists 8,828 links among them.

Comparative approaches

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22741-8_17

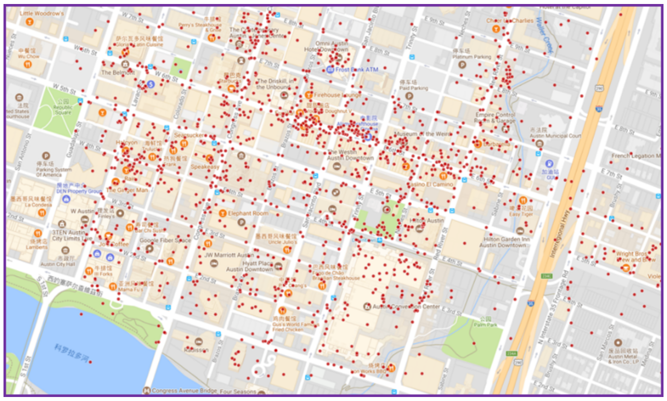


Fig. 5: Locations in the urban area of Austin.

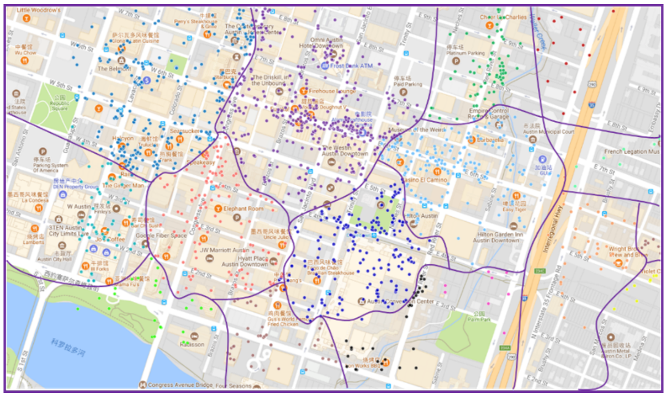


Fig. 6: The visual result of area partition with RST-DBSCAN.

Address method, **Grid** method and **DBSCAN** are taken as comparative approaches in this part.

We obtain regions with these methods at the first step, and compute the interactions of users in these regions to mine unknown social links. S. Scellatopropose, et.al have proved that social links could be mined by user interactions in spatial space and designed several features for mapping relations [15]. We choose **CR** as the feature in this experiment, which is represented as the number of common regions two persons both check in. We calculate CR and assume there exists social links among users whose CR are at the k top. Recall, precision and F1-measure are taken as the metrics to evaluate our approach.

Experimental setup

We set $MinPts = 1$, $\varepsilon = 0.001$ and $\lambda = 3$ for RST-DBSCAN. The $MinPts$ and ε of DBSCAN is the same for a fair comparison. We set gird method with 0.01×0.01 degree, as this is the most effective value and is often used when studying relationships between user interactions in spatial and social link mining.

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22741-8_17

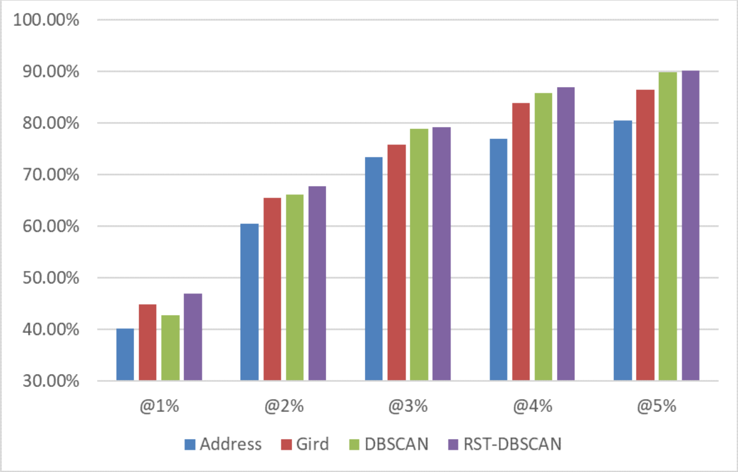


Fig. 7: The recall of social link mining by top-k.

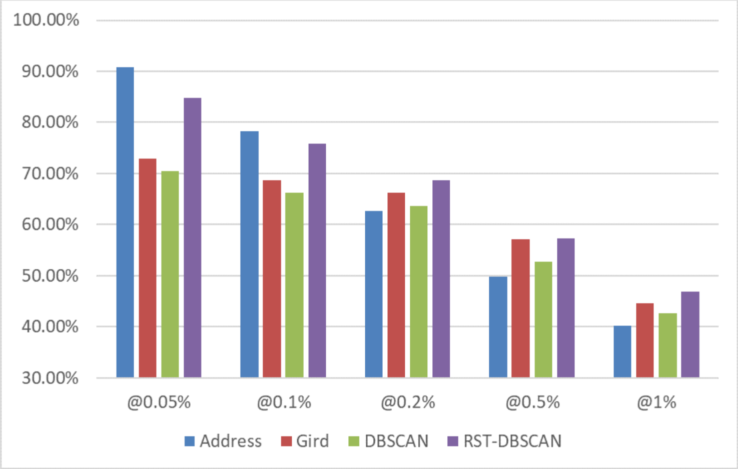


Fig. 8: The precision of social link mining by top-k.

Evaluation results

Results are shown in Fig. 7 and Fig. 8. As can be seen, recall@k of RST-DBSCAN outperforms the other methods, and the advantage of RST-DBSCAN is more obvious when k is smaller. At the same time, precision@k of RST-DBSCAN also outperform that of **Grid** and DBSCAN. An interesting phenomenon is that precision@k of **Address** is the greatest when k is small enough. In fact, it is normal as address is much more precise than other methods, and two persons meet at smaller regions frequently means it is more possible for them to be friends. However, as **Address** method commonly covers small regions, a mass of interactions between two persons are easily missed,

DOI: 10.1007/978-3-030-22741-8_17

leading a rapid drop of Precision@k when k increases. Therefore, compared to others, **Address** is not a stable method when mining social links.

Specially, we compute the F1-measure when k=1, and the values of **Address**, **Grid**, DBSCAN, RST-DBSCAN are 40.14%, 44.72%, 42.71%, 46.95% respectively, and the F1-measure of RST-DBSCAN outperforms others' by 6.81%,2.24%,4.24%. As RST-DBSCAN could partition busy urban areas more reasonably, especially for commercial areas, better results would be obtained when mining social links.

4 Conclusion

In this paper, we present a new area partition method, called RST-DBSCAN, for partitioning busy urban areas with spatial-temporal information in LBSN. Our approach is able to divide the areas into arbitrary and separated regions based on practical significance or independent meaning reasonably. Comprehensive experiments are conducted on a real-world dataset, and results show that our approach performs better than competitors.

References

1. Zheng Yu. *Computing with spatial trajectories*. 2011.
2. Yong Liu, Wei Wei, Aixin Sun, and Chunyan Miao. Exploiting geographical neighborhood characteristics for location recommendation. pages 739–748, 2014.
3. Defu Lian, Yong Ge, Fuzheng Zhang, and N. J Yuan. Content-aware collaborative filtering for location recommendation based on human mobility data. pages 261–270, 2015.
4. Qiang Liu, Shu Wu, Liang Wang, and Tieniu Tan. Predicting the next location: a recurrent model with spatial and temporal contexts. In *Thirtieth AAAI Conference on Artificial Intelligence*, pages 194–200, 2016.
5. Bo Hu and M Ester. Social topic modeling for point-of-interest recommendation in location-based social networks. In *IEEE International Conference on Data Mining*, pages 845–850, 2014.
6. Lars Backstrom, Eric Sun, and Cameron Marlow. Find me if you can: improving geographical prediction with social and spatial proximity. *Www–*, pages 61–70, 2010.
7. Eunjoon Cho, Seth A. Myers, and Jure Leskovec. Friendship and mobility:user movement in location-based social networks. In *ACM SIGKDD International Conference on Knowledge Discovery and Data Mining, San Diego, Ca, Usa, August*, pages 1082–1090, 2011.
8. Huy Pham, Cyrus Shahabi, and Liu Yan. Ebm:an entropy-based model to infer social strength from spatiotemporal data. In *Acm Sigmod International Conference on Management of Data*, 2013.
9. Zhang Kai, Xiaochun Yun, Xiao Yu Zhang, Xiaobin Zhu, Li Chao, and Shupeng Wang. Weighted hierarchical geographic information description model for social relation estimation. *Neurocomputing*, 216:554–560, 2016.
10. Xiao Yu Zhang, Zhang Kai, Xiaochun Yun, Shupeng Wang, and Qingsheng Yuan. Location-based correlation estimation in social network via collaborative learning. In *IEEE INFOCOM 2016 - IEEE Conference on Computer Communications Workshops (INFOCOM WKSHPS)*, 2016.
11. Changqing Zhou, Frankowski Dan, Pamela Ludford, Shashi Shekhar, and Loren Terveen. Discovering personal gazetteers: an interactive clustering approach. In *ACM International Workshop on Geographic Information Systems*, pages 266–273, 2004.

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22741-8_17

12. Changqing Zhou, N Bhatnagar, Shashi Shekhar, and L Terveen. Mining personally important places from gps tracks. In *IEEE International Conference on Data Engineering Workshop*, pages 517–526, 2007.
13. Marc Olivier Killijian. Next place prediction using mobility markov chains. In *The Workshop on Measurement, Privacy, and Mobility*, page 3, 2012.
14. Jiawei Han and Micheline Kamber. *Data Mining: Concepts and Techniques, Second Edition*. China Machine Press, 2006.
15. Salvatore Scellato, Anastasios Noulas, and Cecilia Mascolo. Exploiting place features in link prediction on location-based social networks. In *Acm Sigkdd International Conference on Knowledge Discovery & Data Mining*, 2011.

ICCS Camera Ready Version 2019
To cite this paper please use the final published version:
DOI: 10.1007/978-3-030-22741-8_17